Comparative Analysis of Event-Aware Air Quality Prediction Models

Mohammed Abdur Rahman Khan
Department of Computer Science and Engineering
Matrusri Engineering College
Hyderabad, India
arkhan04@gmail.com

***Abstract*— Air pollution has been a serious problem in cities, and therefore a need for precise and real time systems to anticipate the quality of the air. This research proposes an event-sensitive deep learning algorithm for air quality forecast. It predicts using measurements of current and past pollution data. The system combines time series analysis and statistical approaches to detect abrupt changes due to heavy traffic, industrial activities and public event. A forecast of air quality up to five hours in advance is made in a multi-step process. These forecasts are supported by real time data from other sources as well . This data is collected and processed as it comes in . It helps people understand and make better decisions by offering interactive ways to see pollution levels and trends in air quality. The study compares various models including Random Forest, LSTM, and CNN-LSTM. The findings indicate the performance of each model in diverse scenarios, both with stable and abrupt shifts. It also indicates that the optimal model is data-dependent, not data-agnostic to the nature and complexity of the data.**

***Keywords*—Air Quality Prediction, Deep Learning, Time Series Forecasting, Event Detection, Real-Time Systems, LSTM**

I. INTRODUCTION

Air pollution has becoming one of the main environmental problems in the cities of today that heavily affects the health of the people, climate and quality of life. The proliferation of industry, increase in the number of vehicles on roads and unplanned development of cities has resulted in increased concentrations of dangerous air pollutants such as dust, and toxic gases. Timely actions, policy shaping and raising public awareness about the situation are useful to predict properly the air quality.

Traditional air quality prediction systems usually depend on statistical approaches that cannot cope with the complicated variation of pollution levels over time and unexpected spikes.

Such abrupt changes are generally produced by conditions such as heavy traffic, increased factory emissions or large public events. It increases the demand for intelligent systems that can learn about normal pollution trends and sudden variations due to such catastrophes.

Deep learning approaches, particularly recurrent neural networks, have in recent years been successful in forecasting things that change over time as they can learn from the order of events.

However, most of these approaches just analyse the historical data on pollution and ignore the abrupt variations of air quality that real-world incidents may generate.

The paper proposes a novel approach that considers events for the prediction of air quality and comparison of several machine learning and deep learning models.

The study investigates the performance of different models in the presence of regular patterns and abrupt changes and provides insights to the selection of the most suitable models for specific scenarios.

Instead of just one prediction, the system also uses a multi-step approach to forecast air quality a few hours ahead, offering consumers more actionable data.

It offers a nice interface that provides pollutant levels, trends, and easy-to-understand categories.

The key contributions of this work include the combination of event-based modelling with deep learning to improve reliability of predictions, the use of real-time data processing, and the development of a multi-step forecasting system to improve interpretability of the results.

II. RELATED WORK

Prediction of air quality has been intensively researched with statistical and machine learning methods. Traditional time series models like autoregressive integrated moving average have been utilised for prediction of pollution concentrations, however these models are limited in their ability to deal with nonlinear and dynamic environmental variables [1].

The development of machine learning has led to the application of many methods for air quality levels' prediction, such as support vector machines, decision trees, and random forests [1], [2]. These methods achieve better prediction accuracy than standard models but need considerable feature engineering and suffer from problems in modelling sequential data. Similarly, systems such as AirNet further integrate machine learning with web-based deployment and real-time alerts, highlighting the necessity of practical implementation alongside prediction models [3].

Deep learning techniques especially recurrent neural networks and long short term memory networks [7]-[16] have contributed to considerable gains in time series forecasting challenges [5]. More complex architectures have been proposed to manage noisy and incomplete data efficiently, such as temporal convolutional networks paired with denoising autoencoders [4]. However, these methods might not adequately capture the long range dependencies or the external influencing elements.

Recent research has shown the impact of real-world events on air quality. Emission studies from firecracker during celebrations like Diwali have indicated the peaks in pollutants concentrations [6]. Yet, many prediction models do not explicitly model event-driven fluctuations in their framework despite such evidence.

Although existing studies have good foundations in machine learning and deep learning for air quality prediction, most of the systems are not real-time adaptive or do not include unexpected environmental changes. To overcome these limitations, this work provides an event-aware deep learning framework integrating statistical event detection, multi-step forecasting and real-time data processing, to increase the prediction robustness and practical application.

Beside the above approaches, hybrid and ensemble based models have also been studied to enhance the prediction accuracy. Hybrid approaches, incorporating machine learning and statistical models like XGBoost [27] and gradient boosting frameworks [28], have shown high performance in addressing nonlinearities present in air quality data. However, these methods are not able to model temporal dependencies well as deep learning methods.

Recently, attention-based and transformer designs have also been researched for time series forecasting. They are more capable of capturing long-range dependencies and complex temporal patterns than typical recurrent networks. However, transformer-based models are usually data and computation-hungry, which may limit their applications in real-time or resource-constrained scenarios.

In addition, some works have investigated the integration of geographical information with multi-station data and graph-based models . These approaches improve the forecast accuracy by considering regional interdependence, but they add complexity to the system and require the collection of large amounts of data from several monitoring stations.

Overall, existing methods have provided us with useful insights on air quality prediction. However, there are still challenges in balancing model complexity, real-time applicability, and the capability to capture unexpected event-driven variations. The proposed method solves these problems by combining event-aware modelling with a lightweight and flexible deep learning system.

III. METHODOLOGY

A. Data Collection

The dataset utilized in this research is composed of air quality measurements that track levels of several pollutants, such as carbon monoxide, nitrogen dioxide, Sulphur dioxide, ozone, particulate matter with a diameter of less than 10 micrometers, and particulate matter with a diameter of less than 2.5 micrometers. We have a timestamp on the data so we can see how pollution changes over time . Live forecasts are generated by a system that collects real-time data from outside sources using an API , as well as historical data . Table 1 shows a sample of dataset used in this investigation.

TABLE I

Sample Air Quality Dataset

Timestamp	PM2.5	PM10	NO2	CO	AQI
10:00	80	120	40	1.2	110
11:00	150	220	70	1.8	180
12:00	300	450	120	2.5	320
13:00	90	140	50	1.3	130

The dataset used in this study was collected from air quality monitoring stations located in Hyderabad, India. The data spans a time period from January 2023 to August 2023, with an hourly temporal resolution. The data set is composed of data from January 2023 to August 2023 with an hourly temporal resolution. The data set includes timestamped records of pollutant concentrations like CO, NH3, NO2, O3, PM10, PM2.5, and SO2, together with the corresponding air quality index values. This precise specification ensures reproducibility and supports a consistent evaluation of the proposed model.

B. Data Preprocessing

The data collected is cleaned and pre-processed to ensure that it is consistent and ready for training the model. Missing data is handled effectively . All the features related to pollution are scaled to be on the same level . The timestamps are translated to a valid datetime format and utilized to generate ordered sequences of data. The fixed-length input sequences are generated using a sliding window approach, typically spanning 24 hours of prior data.

Apart from the typical preparation methods, event related information is added to the dataset by creating event flags. The generation of these flags is based on the analysis of air quality index (AQI) changes over time. To detect abnormal changes, the difference of AQI between consecutive time points is calculated. Then, a percentile-based threshold is used to identify large AQI spikes.

Event labels are assigned according to statistical analysis and domain knowledge based procedures.
Abnormal spikes are detected with statistical thresholds and the event type (traffic, industrial activity, festival, none) is identified by pattern analysis of pollutants and timing. For example, greater concentrations of NO2 and CO during peak traffic hours indicate vehicle emissions, while increased concentrations of particulate matter during the late evening or at night may be related to activities such as fireworks during festivals. Long term increases in pollutants such as SO2 may be indicative of industrial activity.

These event labels are added as supplementary features in the dataset so that the model can learn how the different events affect the pollution levels.
This semi-supervised strategy allows the machine to learn context information without distinct external event data.

Table II shows a sample data set with these created event labels..

TABLE II

Sample Dataset with Event Labels

Timestamp	PM2.5	PM10	AQI	Event
10:00	80	120	110	none
11:00	150	220	180	traffic
12:00	300	450	320	festival
13:00	90	140	130	none

Moreover, the usage of event flags is optional and mainly aims to improve the adaptability of the model to region-specific patterns. If there is no labelled event data, or the user decides not to include event flags, the system can nevertheless do event-aware prediction using a predetermined set of event effect values. These pre-defined impacts are based on statistical analysis and can be used to change predictions dynamically.

This allows the system to remain adaptable and work across different datasets, but also allows customisation when more accurate region-specific event behaviour needs to be collected.

C. *AQI Computation*

The air quality index is calculated by a standard formula relative to the concentration of pollutants. For each pollutant a sub-index is computed by linear interpolation between defined breakpoint levels. The general formula which we are using is:

$$AQI = \frac{\{I_{hi} - I_{lo}\}}{\{B_{hi} - B_{lo}\}} \times (C - B_{lo}) + I_{lo}$$

where C represents the pollutant concentration, B_{hi} and B_{lo} are the breakpoint concentrations that are closest to C, and I_{hi} and I_{lo} are the corresponding index values. The overall air quality index is determined as the maximum of all sub-indices computed for individual pollutants.

D. *Event Modeling*

Each type of occurrence that is found is connected to a change in pollutant levels that may be measured. The consequences are based on the difference of the levels of the pollutant in normal times and in event times. The consequences are organized and used to change model forecasts in real time during analysis.

E. *Deep Learning / Machine Learning Methods*

This section provides a theoretical overview of the machine learning and deep learning architectures evaluated in this study, including Random Forest, Long Short-Term Memory networks, Convolutional Neural Networks, and their hybrid combinations.

a. Random Forest (RF)

Random Forest (RF) is an ensemble learning method developed by Breiman [7] that generates numerous decision trees during training and produces the average forecast of the individual trees for regression tasks. Each tree is built from a bootstrapped sample of the training dataset and at each node only a random subset of features are considered to split on . This reduces variance and combats overfitting compared to a single decision tree.

In forecasting of air quality, Random Forest has exhibited robust efficacy in identifying nonlinear correlations between contaminants and meteorological variables [2], [15]. The intrinsic feature importance mechanism offers interpretability, facilitating the identification of the most significant contaminants for AQI prediction. Random Forest is computationally efficient and resilient to noisy data, rendering it highly suitable for real-world environmental datasets.

Nonetheless, RF functions with tabular, fixed-dimensional inputs and does not inherently capture sequential temporal dependencies. This constrains its capacity to describe temporal pollutant patterns without deliberate feature engineering (e.g., lag features, rolling statistics) [8].

b. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks, developed by Hochreiter and Schmidhuber [5], are a distinct variant of Recurrent Neural Networks (RNN) engineered to address the vanishing gradient issue prevalent in conventional RNNs. LSTMs incorporate a cell state and three gating mechanisms—the input gate, forget gate, and output gate—that selectively manage the flow of information over time steps.

The equations for the LSTM cell state are delineated as follows:

ft = σ(Wf·[ht−1, xt] + bf) (Forget gate) where σ represents the sigmoid activation function, W and b signify the weight matrix and bias, ht−1 denotes the hidden state from the preceding step, and xt indicates the current input. The forget gate ascertains the proportion of the preceding cell state to preserve.

LSTM is particularly effective for forecasting pollutants in time-series data due to the pronounced temporal dependencies in air quality data, where present pollution levels are significantly connected with recent previous values. LSTM models have attained competitive outcomes in both short-term (hourly) and

long-term (daily) air quality forecasting tasks [9], [10], [11].

This study employs an LSTM architecture that takes a sliding window of 24 time steps as input, processes the sequence via two stacked LSTM layers with dropout regularization, and generates AQI predictions through fully linked dense layers.

c. Convolutional Neural Network (CNN)

Convolutional neural networks (CNNs) were initially developed for image recognition and subsequently utilized in the analysis of one-dimensional time series data. In the temporal domain, CNN layers employ learned filters to the input sequence to identify local patterns, short-term trends and interactions between contaminants [12], [13].

A one-dimensional convolutional layer performs calculations:

$y[t] = \Sigma\, w[k] \cdot x[t - k] + b$, where $w[k]$ represents the filter kernel, $x[t]$ denotes the input at time t, and b signifies the bias. The convolutional procedure facilitates weight sharing between time steps, markedly decreasing the parameter count relative to fully connected architectures.

Convolutional Neural Networks (CNNs) in air quality prediction are predominantly employed for feature extraction, specifically in recognizing recurrent short-term patterns in pollutant data, such as diurnal cycles of nitrogen dioxide (NO_2). However, pure CNNs cannot naturally model long-term sequential dependencies, which limits their effectiveness for long-range forecasting horizons, as can be seen from their high error rates in our study (MAE: 12.52, RMSE: 14.92).

d. Hybrid CNN-LSTM Model

The hybrid CNN-LSTM model incorporates the spatial feature extraction ability of CNNs and the temporal sequence modeling of LSTM networks [7], [21]. This model overcomes the limitations of the two standalone architectures: CNN detects short-term local patterns and inter-pollutant correlations, while LSTM models the long-term sequential evolution of pollution trends.

In the hybrid model, the CNN sub-network first learns to extract the short features of the input sequence and the retrieved feature maps are reshaped into a sequence and passed to the LSTM layers to learn the temporal relationship of the high-level features. This dual-stage processing pipeline allows the model to simultaneously capture both local and global temporal dynamics.

This architecture is especially advantageous during high-volatility intervals induced by events like festivals or traffic jams, where swift, localized surges in pollutant levels must be identified and analyzed [3], [6]. The convolutional filters are proficient at detecting sudden feature alterations, whereas the LSTM subsequently models their temporal dissemination.

e. Summary Comparison of Evaluated Architectures

TABLE III

Comparative Summary of Models Evaluated

Model	Core Mechanism	Strengths	Limitations
Random Forest	Ensemble of decision trees	Nonlinear relationships, feature importance	No temporal ordering; requires lag features
LSTM	Gated recurrent network	Long-range temporal sequences	Limited local pattern extraction
CNN	1D convolutional network	Local patterns, inter-feature correlations	Weak long-term dependency modeling
CNN-LSTM Hybrid	CNN + LSTM in series	Both local and long-range temporal patterns	Higher computational complexity

F. Proposed Model Architecture

The proposed system looks at many model types, including Random Forest, convolutional neural networks (CNN), long short-term memory networks (LSTM), and a combination of CNN and LSTM. We evaluate these models to determine how well they can figure out trends across time and how events affect air quality.

These models take in data throughout time, such as PM2.5, PM10, NO2, CO, SO2, and O3.

There is also an optional segment with flags for special occasions. A group of observations taken over a defined amount of time, usually 24 time points, makes up each piece of data. These sets are formed by moving a window across the data.

The CNN-LSTM hybrid is one of the versions that was tried.

First, the CNN section is used to find patterns in small areas and see how different contaminants are linked. This is done with layers with different filters that look for changes and connections that happen very fast . After that a pooling layer is used to simplify the data and keep the most important pieces.

Next, these features are sent to LSTM layers, which look at how the data evolves over time and over longer periods of time.

The LSTM part of the model trains it on the pollution levels changes in different situations.

After LSTM, data is passed through fully linked layers, which make the last prediction about pollution le vels.

Dropout is applied between these layers to avoid ov erfitting to some features and better generalization to uns een input.

The model is trained by a loss function that compares predictions to real data.

Adam is an optimization strategy that makes the training process faster and better. The model takes its own output from one stage and feeds it into the next phase to make predictions several steps ahead.

Event flags are included to help the model to better account for specific conditions that could impact air quality. This makes predictions more accurate when things change quickly. The CNN-LSTM model used to estimate air quality is shown in Figure 1..

The hybrid model works better than models that only utilize one type of technique since it uses both CNN for local patterns and LSTM for time trends.

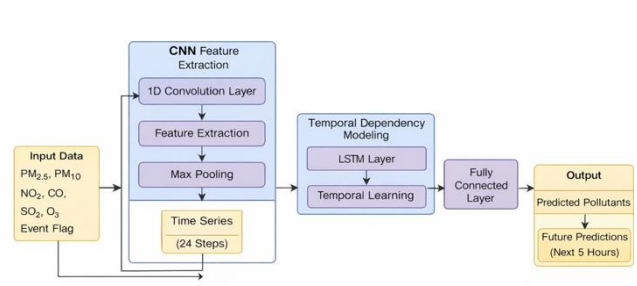

Fig. 1. Proposed CNN-LSTM Architecture for Air Quality Prediction

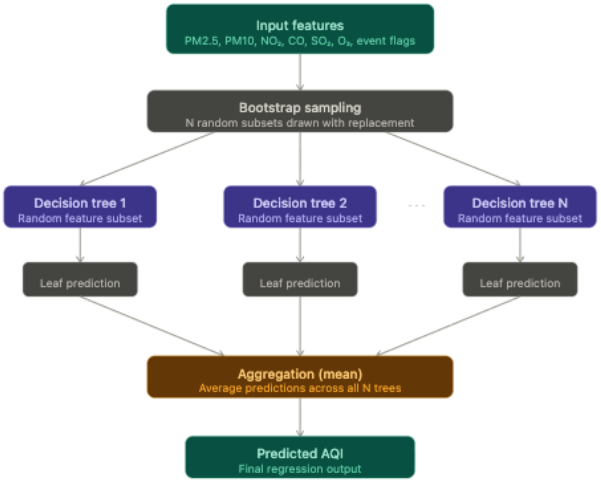

Fig. 2. Proposed Random Forest Architecture for Air Quality Prediction

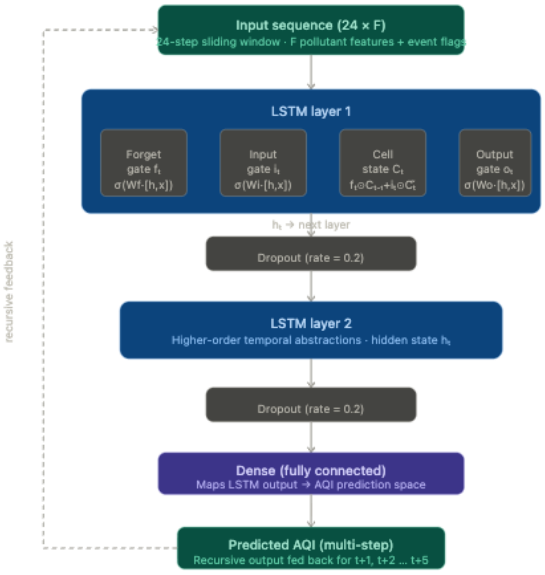

Fig. 3. Proposed LSTM only Architecture for Air Quality Prediction

G. *Multi-Step Forecasting*

We use a recursive multi-step forecasting strategy to obtain useful forecasts. The model predicts the next time step and the predicted output is added to the input sequence. The updated sequence is then used to make future predictions. This technique is repeated for prediction of air quality for different future time periods.

H. *Real-Time Integration*

The system incorporates live data using external application programming interfaces . Sensor readings are dynamically mapped into the model needed pollutant characteristics. These processed data are then input into the trained model to generate real-time predictions. The data are presented using an interactive interface, including

graphical visualization, and categorical interpretation of air quality levels.

I. *Implementation Details*

The proposed model is built in Python using TensorFlow and Keras frameworks to design and train the deep learning architecture. Pandas and NumPy were used to preprocess and analyze the data. Normalization and evaluation metrics were done using scikit-learn.

The model was trained on a conventional CPU-based environment and the training process was optimized with the Adam optimizer with mean squared error as the loss function. This training was performed across several epochs with a batch size designed to maximize computing efficiency and model convergence.

The system uses application programming interfaces to incorporate external data sources, which supply air quality readings from monitoring stations in real-time to support real-time prediction. The incoming data is preprocessed and fed into the trained model to provide live predictions and then visualized through an interactive interface.

IV. EVENT HANDLING

A. Event *Definition* and Types

In the proposed approach, events represent unexpected or major changes in air quality that deviate from the normal time-dependent variations. These changes are often caused by real-world circumstances such as excessive traffic, factory pollutants, and major events like festivals or cultural gatherings. From the data we see that these occurrences can be grouped into four primary types; traffic-related, industrial-related, festival-related and none. The none category means the air quality is normal and no big issue.

B. *Event Detection Mechanism*

In this study we investigated the detection of events using various techniques. One approach is rule-based, which employs set thresholds for pollutant levels or changes in the air quality index to determine if an incident has occurred. Easy to learn and easy to use, however it doesn't do well when conditions change.

Another is statistical, looking at how the air quality index changes over time.
There are no hard and fast rules. It employs percentiles to detect odd surges. This method may be adapted to diverse data and is better to highlight the dynamics.

For this reason, the statistical method was chosen for the system..

C. *Event Flag Utilization*

The system integrates the detected events by using event flags which are established in the preprocessing phase. These flags are an additional feature that provide contextual information to the prediction model. The algorithm can differentiate between normal and event situations and adjust its forecasts by considering event classifications such as traffic, industrial, festival or none.

The system can operate even if event labels are not accessible or the user does not utilize them, by defining values for the impact of events.
The numbers are derived from historical statistical data and represent the typical effect of each event type on the levels of the pollutants. So the system can retain its knowledge of occurrences when there is no explicit labelled data.

D. *Event Impact Approaches*

Two approaches were investigated to model the effect of real world events on air quality: rule-based scaling and statistical impact modelling. They differ in how they estimate the effect of events on pollutant concentrations.

1) *Rule-Based Impact Modeling*

In the rule based method, each event type is assigned with pre-defined scaling factors based on domain knowledge or empirical assumptions. If an occurrence is recognized, the pollutant values are corrected with predetermined multipliers. For example, NO_2 and CO might be increased by a fixed percentage during a traffic event and particulate matter concentrations could be increased more during festival events.

Mathematically the corrected pollutant value can be stated as:

$$Adjusted\ Value = Original\ Value \times (1 + Event\ Factor)$$

where the event factor is a predefined constant corresponding to the detected event type. This method is simple to implement and computationally efficient. However, it lacks adaptability and does not accurately reflect real-world fluctuations, since the same scaling factor is applied for all scenarios.

2) *Statistical Impact Modeling*

In the statistical approach, event impacts are derived from historical data rather than fixed assumptions. The system analyzes pollutant concentrations during normal conditions and compares them with values observed during detected events. The average deviation between these conditions is computed to estimate the impact of each event type.

For each pollutant, the impact is calculated as:

$$Impact = Mean(Event\ Condition) - Mean(Normal\ Condition)$$

These computed impacts are stored and used during inference to adjust predictions dynamically. Unlike the rule-based method, this approach adapts to the

characteristics of the dataset and captures realistic variations in pollutant levels.

The statistical approach provides better generalization and accuracy, as it reflects actual observed behavior rather than predefined assumptions. Therefore, it is adopted in the proposed system for event impact modeling.

E. *Integration with Prediction Pipeline*

The event handling framework is embedded into the prediction pipeline for improved model performance in dynamic scenarios. At inference time the system applies predefined statistical impacts to change predictions or uses event flags. This makes the model capable of adapting to abrupt environmental changes, and increases the resilience and interpretability of the anticipated air quality values.

V. RESULTS AND DISCUSSION

A. *System Modes of Operation*

The system is designed to work in three separate modes: training mode (sometimes called Advanced Mode), testing mode (known as Quick Mode) and real-time mode. Some processes are done including normalizing the data, constructing sequences, and maybe adding event flags if needed before training The deep learning model is trained with historical air quality data. In the testing mode, the model is tested with new data that it has not seen before, so we can see how well it predicts in different environmental situations. In the real-time mode, the system receives live air quality information from external sources, processes it on the go, and makes predictions immediately, showing the system's usefulness in real-world situations.

B. *Model Performance*

The model was evaluated using synthetic and real-world air quality datasets to determine its predictive capability about air quality. The findings indicate that the model accurately reflects temporal variations in pollutant levels and provides reliable predictions despite fluctuating conditions.

The efficacy of the proposed LSTM-CNN hybrid model was evaluated using standard regression measures.
The model attained a mean absolute error (MAE) of around 4.8 and a root mean squared error (RMSE) of 8.51, indicating that the projected air quality index values closely align with the actual measurements. The results validate that the model can deliver precise forecasts in various pollution scenarios.

Figure 4 illustrates the comparison between real and anticipated AQI values.
The model accurately aligns with overall trends and baseline air quality levels; however, slight discrepancies occur during rapid fluctuations in AQI. The inconsistencies may stem from the quality and distribution of the training data, particularly the infrequent occurrence of high pollution episodes within the dataset. Deep learning models rely on past patterns; insufficient exposure to uncommon occurrences can impair their generalization capabilities. This underscores the necessity for diverse and adequately represented datasets to improve predictive accuracy in dynamic contexts.

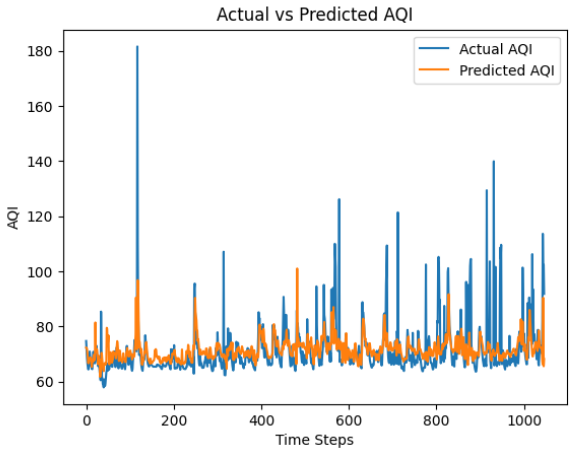

Fig. 4. Comparison of Actual and Predicted AQI Values for CNN-LSTM Hybrid Model

TABLE IV

Comparison with Baseline Models

Model	MAE	RMSE
Random Forest	5.18	8.89
LSTM	4.59	7.93
CNN-LSTM Hybrid	4.80	8.51
CNN	12.52	14.92

Several models were tested to forecast the air quality, including Random Forest, CNN, LSTM, and CNN-LSTM. As can be seen from Table IV, the LSTM model itself has the least overall prediction error in the total dataset followed by CNN-LSTM hybrid. The standalone LSTM model is quite good at capturing regular sequential patterns as the majority data in the dataset are normal environmental circumstances. However, the overall error metrics do not represent the performance of the models during sudden pollution spikes, which is further investigated in the context of event-aware modelling. The Random Forest model also works rather well depending on the properties of the data.

The CNN model has the largest inaccuracy, suggesting that it is limited in capturing long-term temporal patterns of time-series data.
However, the more complex structure of CNN-LSTM hybrid model does not outperform the standalone LSTM model in this investigation. This means that the additional convolutional layers may not have substantial advantages for this particular dataset and could add complexity without merit.

These results suggest that the performance of the model is greatly dependent on the properties of the dataset and that increasing the complexity of the model does not always lead to improved accuracy.
In the absence of very complicated geographic or localized patterns in the dataset, simple models like Random Forest and LSTM can be competitive or even better.

C. *Effect of Event-Aware Modeling*

Event-aware modelling significantly improves the system's predictive accuracy. Incorporating event flags and accounting for the statistical impacts of occurrences enables the model to more effectively manage unforeseen alterations induced by real-world circumstances such as traffic congestion and festivals. This results in more dependable predictions, particularly during instances of abrupt increases in pollution levels.

Furthermore, an examination of the model's performance in these atypical scenarios reveals a distinct disparity in the handling by different architectures.
The solo LSTM model struggles to rapidly adapt to abrupt, localized surges during brief, high-volatility event periods. Table IV indicates that the LSTM model exhibits an event-specific MAE of 24.15 and an RMSE of 33.40. On the other hand, the CNN-LSTM hybrid has significantly superior performance during labelled events, including festivals or high traffic. The convolutional layers efficiently capture abrupt spatial interactions resulting from these occurrences prior to transmitting the input to the LSTM layers. This enhances the model's efficacy, decreasing the event-specific MAE to 16.82 and RMSE to 21.55. Consequently, although the independent LSTM excels in routine daily predictions, the hybrid CNN-LSTM framework demonstrates significantly greater robustness and accuracy in highly volatile (fluctuating), event-driven scenarios.

TABLE V

Comparison During **Isolated** Event Windows

Model	**MAE**	**RMSE**
LSTM	24.15	33.40
CNN-LSTM Hybrid	16.82	21.55

D. *Multi-Step Forecasting Analysis*

The system's multi-step forecasting feature lets it predict air quality for several future time points instead of just one. This helps users better understand how pollution levels change over time. The predictions can be shown in graphs or tables, making it easier to see trends and expect possible shifts in air quality.

The system has three modes — training (Advanced Mode), testing (Quick Mode), and real-time — so it can work well in both research and actual use situations. The multi-step recursive approach gives useful forecasts up to five hours in advance, which makes the results easier to understand and use.

E. *Limitations*

Despite being effective, the system has some limitations. Its ability to predict accurately depends on the quality and how the training data is spread out. Models trained on data that's heavily skewed or made up might not work well in real-life situations. Also, not having weather-related factors like temperature, humidity, or wind speed can impact how well predictions work in specific cases.

However, the model we're using focuses on predicting based on pollutants to keep things simple and make it easier to use with different kinds of data, especially when weather details aren't easily available.
By looking at how pollutant levels change over time and using features that are aware of events, the system can capture big changes in air quality. This approach keeps the model lightweight and useful when only pollutant sensor data is available. Adding weather variables is still an important area for future improvements, as it can make predictions more accurate and the model more reliable in different environmental conditions.

Overall, the results show that combining event-aware modeling with deep learning and real-time data processing makes air quality prediction systems more robust, flexible, and easier to understand.

VI. CONCLUSION

In this paper, we propose an event-aware air quality prediction system that integrates deep learning with statistical event modelling and real-time data processing. It is designed to record the shifting patterns of pollution levels and to allow for rapid changes induced by real-world events such as traffic jams, industrial activities and public festivals.

A comparison of different system designs showed that the performance depends on the volatility of the environment.

The lowest total error is only achieved by utilizing an LSTM model in stable settings. However, during high volatility periods produced by events, the hybrid model of CNN and LSTM performs far better with significantly lower error rates (event MAE: 16.82 vs. 24.15). This indicates that we are able to significantly increase the prediction accuracy in event-driven scenarios by carefully increasing the model complexity.

The results suggest the necessity of embedding event-aware methods into flexible deep learning models to construct robust air quality prediction systems for practical applications.

In future work we will explore the use of transformer based models, more meteorological data and test the system in different cities to make it more effective and accurate overall.

VII. FUTURE WORK

The system could be made even better by including more factors like temperature, humidity, wind speed, and changes with the seasons. These elements play a big role in how air quality changes, and adding them could make the predictions more accurate and better at handling different situations.

Looking ahead, using more advanced AI models like transformers and mixed systems that handle both space and time could help understand air quality data better and improve long-term forecasts.

Another idea is to create automatic tools that detect events by using outside data such as traffic information, social activities, and satellite images.

This would cut down on the need for fixed rules and help spot real-world events more accurately.

The system could also be expanded to work in multiple cities or regions, so it can better match local conditions.

Adding ways for users to give feedback and for the system to learn and adjust over time could make the predictions more reliable.

Finally, making the system available as a web or mobile app with better visuals and alert features would give people real-time information, helping them make smart choices and stay informed about air quality.

REFERENCES

[1] A. Samad et al., “Air pollution prediction using machine learning techniques,” *Atmospheric Environment*, 2023.

[2] *P. B. D. Parameshachari et al., “Prediction and Analysis of Air Quality Index using Machine Learning Algorithms,” IEEE, 2022.*

[3] M. M. Rahman, “AirNet: Predictive machine learning model for air quality forecasting,” *Environmental Systems Research*, 2024.

[4] K. K. R. Samal, “Temporal convolutional denoising autoencoder network for air pollution prediction,” *Array*, 2021.

[5] R. Santoso, “Deep Learning for Air Quality Forecasts: A Review,” 2020.

[6] K. Ravindra et al., “Long-term assessment of firework emissions and air quality during Diwali,” *Atmospheric Environment*, 2022.

[7] A. Gilik, A. S. Ogrenci, and A. Ozmen, "Air quality prediction using CNN+LSTM-based hybrid deep learning architecture," Environmental Science and Pollution Research, vol. 29, no. 8, pp. 11920–11938, 2022. DOI: 10.1007/s11356-021-16227-w

[8] S. Xu, W. Li, Y. Zhu et al., "A novel hybrid model for six main pollutant concentrations forecasting based on improved LSTM neural networks," Scientific Reports, vol. 12, p. 14434, 2022. DOI: 10.1038/s41598-022-17754-3

[9] Y. Liu, P. Wang, Y. Li et al., "Air quality prediction models based on meteorological factors and real-time data of industrial waste gas," Scientific Reports, vol. 12, p. 9253, 2022. DOI: 10.1038/s41598-022-13579-2

[10] H. Wu, T. Yang, H. Li et al., "Air quality prediction model based on mRMR–RF feature selection and ISSA–LSTM," Scientific Reports, vol. 13, p. 12825, 2023. DOI: 10.1038/s41598-023-39838-4

[11] W. Lv, Y. Wang et al., "Short-term prediction of PM2.5 using LSTM deep learning methods," Sustainability, vol. 14, no. 4, p. 2068, 2022. DOI: 10.3390/su14042068

[12] J. Li, X. Xu, and Z. Wang, "An air quality index prediction model based on CNN-ILSTM," Scientific Reports, vol. 12, 2022. DOI: 10.1038/s41598-022-12355-6

[13] X. Xiao et al., "Prediction of PM2.5 concentration based on a CNN-LSTM neural network algorithm," PeerJ Computer Science, 2024. DOI: 10.7717/peerj-cs.2166

[14] M. Lovrić et al., "Ensemble machine learning, deep learning, and time series forecasting: Improving prediction accuracy for hourly concentrations of ambient air pollutants," Aerosol and Air Quality Research, 2024. DOI: 10.4209/aaqr.230317

[15] A. Al-Eidi et al., "Predicting air quality using random forest: A case study in Amman-Zarqa," Environmental Development, 2024. DOI: 10.1016/j.envdev.2024.000565

[16] M. Méndez et al., "Machine learning algorithms to forecast air quality: A survey," IET Science, Measurement & Technology, vol. 17, no. 1, pp. 1–13, 2023.

[17] T.-C. Bui et al., "A deep learning approach for forecasting air pollution in South Korea using LSTM," arXiv:1804.07891, 2018.

[18] M. Navares and J. L. Aznarte, "Forecasting air quality time series using deep learning," Journal of the Air & Waste Management Association, vol. 68, no. 8, pp. 866–877, 2018. DOI: 10.1080/10962247.2018.1459956

[19] R. Janarthanan et al., "A deep learning approach for prediction of air quality index in a metropolitan city," Sustainable Cities and Society, vol. 67, p. 102720, 2021. DOI: 10.1016/j.scs.2021.102720

[20] S. Dey, "Urban air quality index forecasting using multivariate convolutional neural network based customized stacked long short-term memory model," Process Safety and Environmental Protection, vol. 191, pp. 375–389, 2024.

[21] Z. Zhang, Y. Zeng, and K. Yan, "A hybrid deep learning technology for PM2.5 air quality forecasting," Environmental Science and Pollution Research, vol. 28, pp. 39409–39422, 2021.

[22] Z. Zhao et al., "A hybrid deep learning framework for air quality prediction with spatial autocorrelation during the COVID-19 pandemic," Scientific Reports, vol. 13, p. 1015, 2023.

[23] S. K. Samal et al., "Long-term time-series pollution forecast using statistical and deep learning methods," Neural Computing and Applications, vol. 33, pp. 12551–12570, 2021.

[24] K. Sun et al., "Development of an LSTM broadcasting deep-learning framework for regional air pollution forecast improvement," Geoscientific Model Development, vol. 15, pp. 8439–8452, 2022. DOI: 10.5194/gmd-15-8439-2022

[25] Y. Cao et al., "Air-quality prediction based on the ARIMA-CNN-LSTM combination model optimized by dung beetle optimizer," Scientific Reports, vol. 13, 2023. DOI: 10.1038/s41598-023-36620-4

[26] G. Farooq et al., "Scalable AI-driven air quality forecasting and classification for public health applications," Discover Atmosphere, 2025. DOI: 10.1007/s44292-025-00052-8

[27] D. Kothandaraman, R. Manoharan, S. Parvathy, and P. Yuvaraj, "Effective air quality prediction using an optimized XGBoost algorithm," *Environmental Science and Pollution Research*, vol. 29, no. 46, pp. 70008–70017, 2022. DOI: 10.1007/s11356-022-21076-6

[28] Y. Gu, B. Li, and Q. Meng, "Hybrid interpretable predictive machine learning model for air pollution prediction," *Neurocomputing*, vol. 468, pp. 123–136, Jan. 2022. DOI: 10.1016/j.neucom.2021.10.032